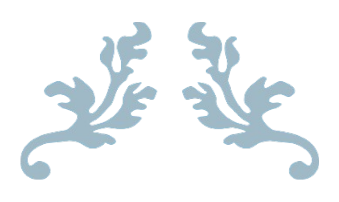


TRABAJO TEÓRICO TECO

TECNOLOGÍA DE COMPUTADORES



19 DE ENERO DE 2025

UCLM

BALTER MANEL CASTILLO CUEVAS
RUTH FERNANDEZ NAVARRO
MARIO SIERRA INIESTA
ALEJANDRO VICIOSO OVIEDO

Contenido

Introducción

En este presente documento explicaremos las funciones de un circuito dado donde cada miembro posee un circuito único de que da una secuencia dada por el DNI/NIE de los integrantes, además de la implementación de estas utilizando un decodificador, multiplexores y puertas lógicas en cada biestables, el primer circuito contara con 2 subcircuitos donde el primer subcircuito realizara la secuencia mencionada y el segundo subcircuito contara las cantidad de veces que se cumplió la secuencia correcta.

Se explicará además el uso de cada una dependiendo de la tabla de la verdad que se entrega junto este documento para comprobar la secuencia correcta.

Explicación teórica del caso general a resolver

El problema por resolver se puede dividir en cuatro partes:

1. Creación de la tabla de la verdad a partir de la que se codificarán los estados y se diseñará el circuito con el objetivo de diseñar un circuito que repita una secuencia de 5 números diferentes el que variará en función de cada miembro del grupo
2. Creación del primer subcircuito, a partir de la tabla se implementarán los biestables atendiendo a las condiciones puestas para crear un circuito que repita la secuencia deseada, siendo las condiciones utilizar un decodificador 2x4 para T3 y T2, implementar solo D1 a través de puertas lógicas y J0 K0 con un multiplexor 8x1
3. Creación del segundo subcircuito, este subcircuito consiste en crear un contador de 0 a 3 para contar las veces que se repite la secuencia de cada uno
4. Main, se implementarán ambos subcircuitos donde se demostrará el correcto funcionamiento del programa, para ello se utilizarán dos Hex Digit Display para mostrar en uno la repetición de la secuencia de números y en el otro las veces que este repite

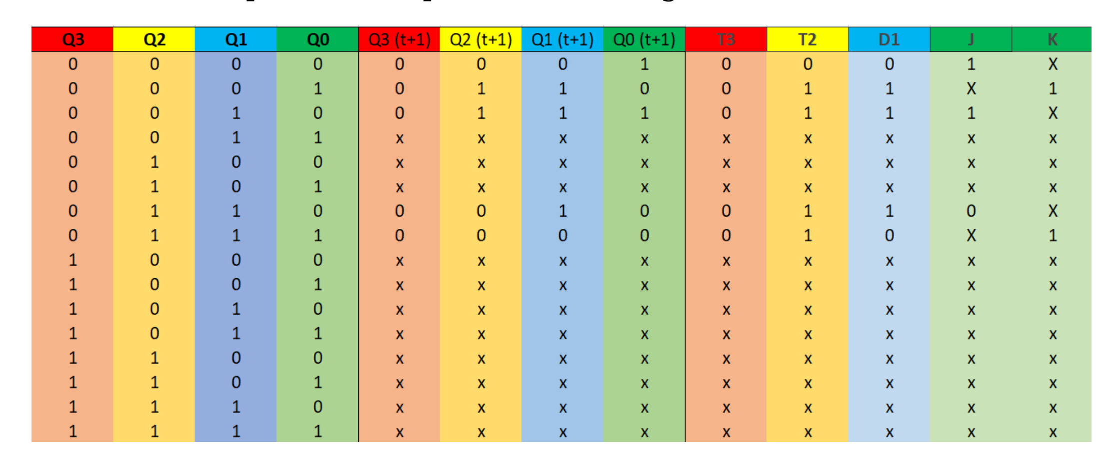
Desarrollo personal de la práctica

Balter Manuel Castillo Cuevas

Subcircuito1:

Identifico mi secuencia a través de las instrucciones dadas y me encuentro con la 2-7-0-1-6, una vez identifico la secuencia determino los bits a utilizar en este caso utilizare 4 bits, aunque los números 8 y 9 no se utilizaran, pero el ejercicio solicita el uso de los 4 biestables, por lo que se realizara de igual manera con 4 bits, además determinamos el uso de los biestables que nos solicita el ejercicio que será los biestables T2, D1 y J0K0.

Una vez determinado eso procedemos a realizar la tabla de estados con la secuencia correspondiente, quedaría de la siguiente manera:

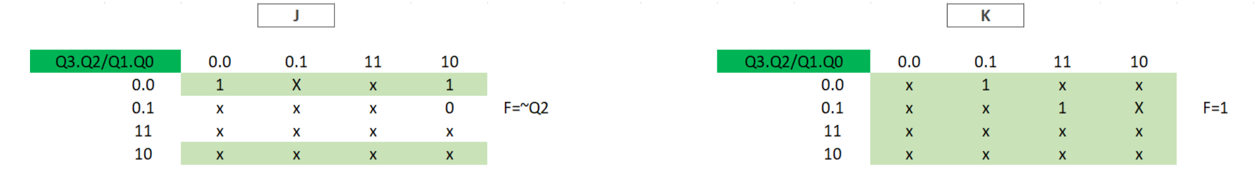


Q3	Q2	Q1	Q0	Q3 (t+1)	Q2 (t+1)	Q1 (t+1)	Q0 (t+1)	T3	T2	D1	J	K
0	0	0	0	0	0	0	1	0	0	0	1	X
0	0	0	1	0	1	1	0	0	1	1	X	1
0	0	1	0	0	1	1	1	0	1	1	1	X
0	0	1	1	x	x	x	x	x	x	x	x	x
0	1	0	0	x	x	x	x	x	x	x	x	x
0	1	0	1	x	x	x	x	x	x	x	x	x
0	1	1	0	0	0	1	0	0	1	1	0	X
0	1	1	1	0	0	0	0	0	1	0	X	1
1	0	0	0	x	x	x	x	x	x	x	x	x
1	0	0	1	x	x	x	x	x	x	x	x	x
1	0	1	0	x	x	x	x	x	x	x	x	x
1	0	1	1	x	x	x	x	x	x	x	x	x
1	1	0	0	x	x	x	x	x	x	x	x	x
1	1	0	1	x	x	x	x	x	x	x	x	x
1	1	1	0	x	x	x	x	x	x	x	x	x
1	1	1	1	x	x	x	x	x	x	x	x	x

Se nos solicita en el primer subcircuito realizar la secuencia, aclarada anteriormente utilizando los biestables correspondientes y siguiendo las instrucciones que nos solicitan para realizar el ejercicio.

Biestable J0K0: Para este biestable es necesario utilizar un multiplexor 8x1, por lo que siguiendo nuestra tabla nos dará que J0 y K0 controlan un biestable que participa en la secuencia según el circuito, K0 es constante y vale 1, lo que simplifica su funcionamiento e implementamos el multiplexor de manera que se utiliza para manejar las entradas al biestable J0, determinando su comportamiento en función de múltiples condiciones de control. Las líneas de control del MUX (Q3, Q2, Q1) dividen la tabla en 8 casos posibles, permitiendo seleccionar la entrada correcta en cada momento.

Una vez determinado la tabla de la verdad realizamos el uso del Mapa de Karnaugh tanto para el J0 Y K0:



J

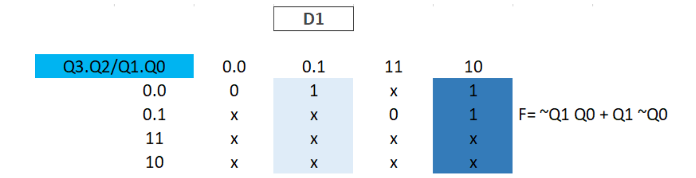
Q3.Q2/Q1.Q0	0.0	0.1	11	10	
0.0	1	X	x	1	
0.1	x	x	x	0	F=~Q2
11	x	x	x	x	
10	x	x	x	x	

K

Q3.Q2/Q1.Q0	0.0	0.1	11	10	
0.0	x	1	x	x	
0.1	x	x	1	X	F=1
11	x	x	x	x	
10	x	x	x	x	

Biestable D1: En este caso nos solicita que la implementación sea con puertas lógicas por lo que siguiente la tabla de la verdad debemos implementarla con mapa de Karnaugh para simplificarla lo más posible, la tabla de la verdad nos queda de la siguiente manera:

Una vez obtenida la tabla de verdad procedemos a utilizar mapas de Karnaugh para su simplificación como se había mencionado anteriormente, queda la simplificación de la siguiente manera:



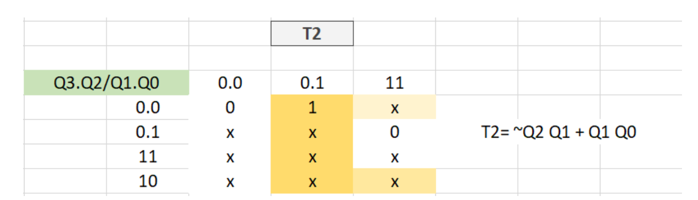
D1

Q3.Q2/Q1.Q0	0.0	0.1	11	10
0.0	0	1	x	1
0.1	x	x	0	1
11	x	x	x	x
10	x	x	x	x

F= ~Q1 Q0 + Q1 ~Q0

Biestable T2: Se implementa en este biestable lo que nos solicita el ejercicio que es un decodificador 2x4, que depende la salida tanto de la Q3 como de la Q2, todo eso implementado para utilizarlo con las puertas lógicas correspondientes que identificamos a través de la tabla de la verdad correspondiente:

Una vez obtenida la tabla de verdad procedemos a utilizar mapas de Karnaugh para su simplificación como se había mencionado anteriormente, queda la simplificación de la siguiente manera:

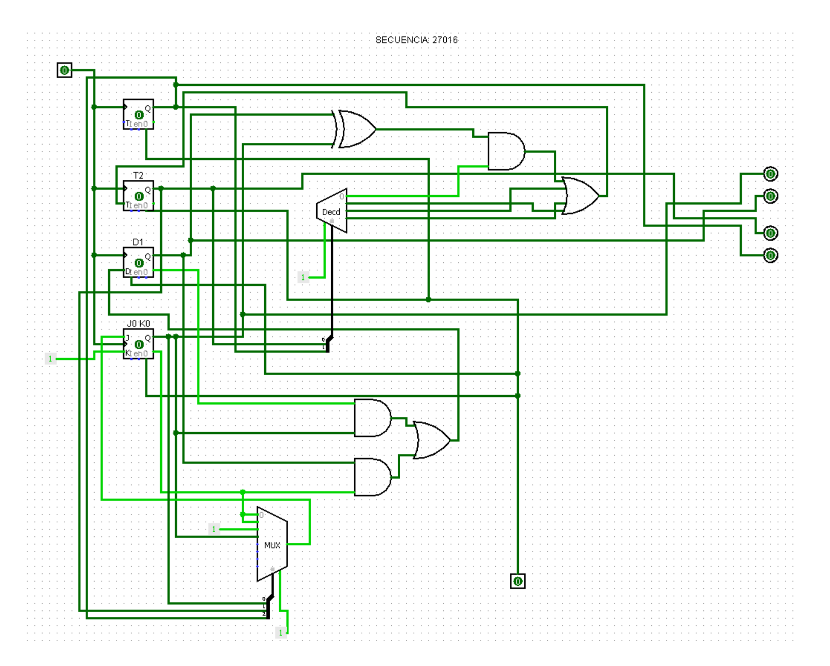


T2

Q3.Q2/Q1.Q0	0.0	0.1	11
0.0	0	1	x
0.1	x	x	0
11	x	x	x
10	x	x	x

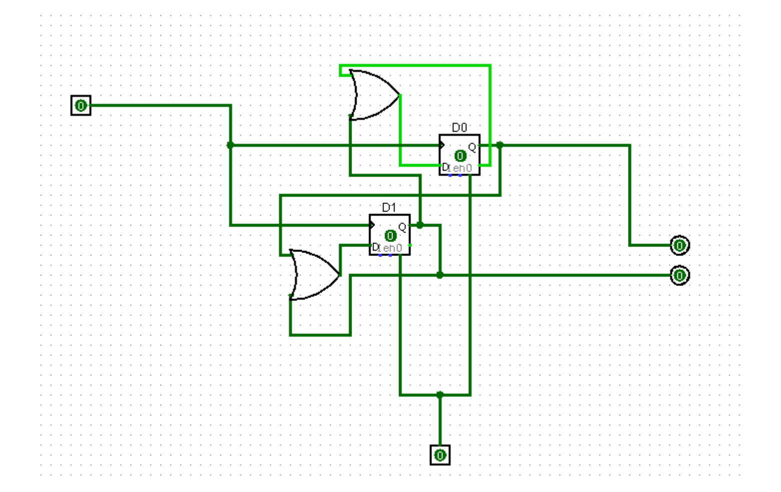
T2= ~Q2 Q1 + Q1 Q0

Biestable T3: En este caso el T3 no correspondería con ningún numero de la secuencia por lo que procedemos a implementar las salidas en 0 y se cancelaría este biestable, porque no utilizaremos el decodificador 2x4 en este caso para el mismo, nos queda la siguiente tabla de la verdad en todo caso. El primer subcircuito quedaría de la siguiente manera, implementando todas las tablas de verdad anteriormente mencionadas:



Subcircuito2:

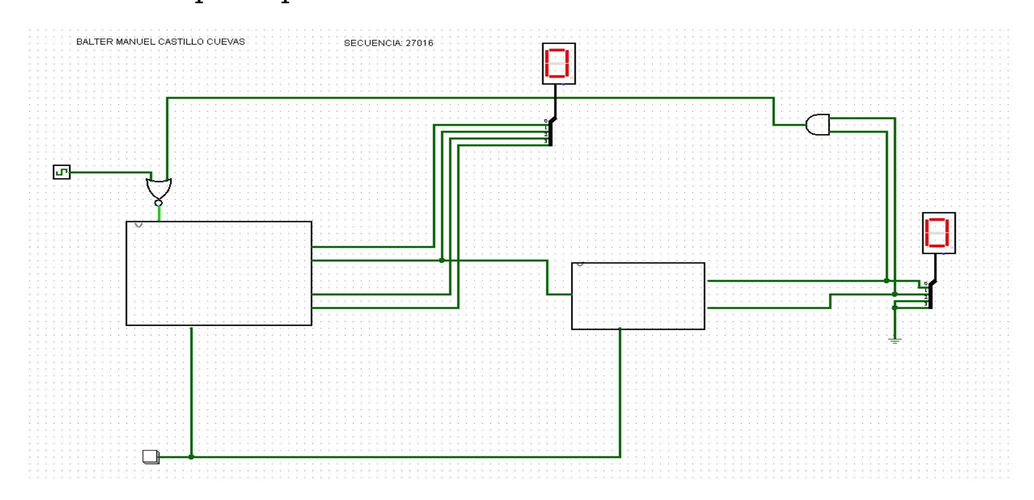
Para el subcuito 2 debemos implementar un contador de 2 bits que realice la cuenta desde 0 hasta el 3 del primer subcircuito por lo que una vez que termine la cuenta debería de detenerse el primer subcircuito, para ellos nos guiamos de una las practicas solicitadas del laboratorio. ~~En~~ este circuito, el reloj se conecta al primer biestable a través de una entrada variable que se activa cuando un comparador detecta que el número en el que se encuentra la secuencia del subcircuito 1 corresponde al primer número de la secuencia. Esto permite que, tras detectar este número tres veces (sin contar la inicialización, ya que el contador empieza en 0), el contador emita una señal conectada a los *clear* y *preset* de los biestables del subcircuito 1, fijando su estado en el primer número de la secuencia. En este caso, el estado final queda en el número 1.



Circuito MAIN:

En cuanto al main, simplemente integramos los subcircuitos 1 y 2, estableciendo las conexiones necesarias entre ellos para cumplir con el

objetivo del apartado. Además, he vinculado cada subcircuito a un display HEX mediante divisores de 4 bits, lo que permite observar en tiempo real el número que representa cada secuencia en cada momento.



Ruth Fernández Navarro

Mi secuencia es: 4-2-6-0-5, 4-2-6-0-5, 4-2-6-0-5, 4…

Para implementarla utilizaré 3 bits, representando todos los números de la secuencia. Al usar 3 bits, solo son necesarios utilizar 3 biestables. Se pide usar 4, pero debido a los números de mi secuencia, con 3 es suficiente (T3 sería 0, y el decodificador unido a él no tendría salidas).

De este modo usaré los biestables T2, D1 y J0K0, como se especifica en los requisitos para completar la siguiente tarea.

Esta es la tabla a partir de la cuál he obtenido todos los datos necesarios para llevarla a cabo:

Q3	Q2	Q1	Q0	Q3(t+1)	Q2(t+1)	Q1(t+1)	Q0(t+1)	T2	T3	D1	J0	K0
0	0	0	0	0	1	0	1	1	0	0	1	x
0	0	0	1	x	x	x	x	x	x	x	x	x
0	0	1	0	0	1	1	0	1	0	1	0	x
0	0	1	1	x	x	x	x	x	x	x	x	x
0	1	0	0	0	0	1	0	1	0	1	0	x
0	1	0	1	0	1	0	0	0	0	0	x	1
0	1	1	0	0	0	0	0	1	0	0	0	x
0	1	1	1	x	x	x	x	x	x	x	x	x
1	0	0	0	x	x	x	x	x	x	x	x	x
1	0	0	1	x	x	x	x	x	x	x	x	x
1	0	1	0	x	x	x	x	x	x	x	x	x
1	0	1	1	x	x	x	x	x	x	x	x	x
1	1	0	0	x	x	x	x	x	x	x	x	x
1	1	0	1	x	x	x	x	x	x	x	x	x
1	1	1	0	x	x	x	x	x	x	x	x	x

1	1	1	1	x	x	x	x	x	x	x	x	x

Lo mostrado en la tabla, desde la izquierda, son los estados necesarios (Q3, Q2, Q1 Y Q0). Los estados posteriores son los que vienen más a la derecha, serían (Q3(t+1), Q2(t+1), Q1(t+1) y Q0(t+1)). Lo que les siguen son los biestables necesarios (ahí se observa por qué T3 es 0), que cambian los estados a los números de la secuencia, y provienen de la tabla de excitación.

CIRCUITOS Y SU IMPLEMENTACIÓN

1. SUBCIRCUITO 1.

El primer subcircuito es el encargado de que se repitan los números de la secuencia indefinidamente, en el orden indicado. Para hacerlo, las instrucciones del ejercicio pedían eran que:

- T2 Y T3 fueran implantados con decodificadores 2x4.
- D1 fuese implantado con puertas lógicas (sin especificar).
- J0K0 fuese implantado con multiplexores 8x1.

PARA T3: Como he mencionado anteriormente, las salidas serían 0 y se cancelaría este biestable. Las variables de control serían Q2 y Q3, en ese orden de prioridad de arriba a abajo.

Decodificador 2X4 T3 2^2

Q3(t)	Q2(t)	Q1(t)	Q0(t)	T3	S
0	0	0	0	0	0
0	0	0	1	x	
0	0	1	0	0	
0	0	1	1	x	
0	1	0	0	0	0
0	1	0	1	0	
0	1	1	0	0	
0	1	1	1	x	
1	0	0	0	x	0
1	0	0	1	x	
1	0	1	0	x	
1	0	1	1	x	
1	1	0	0	x	0
1	1	0	1	x	
1	1	1	0	x	
1	1	1	1	x	

PARA T2: En este caso, sí que habría que decidir cuáles son las variables de control y las salidas dependiendo de los datos en la tabla. Las variables elegidas serían Q2 y Q3, como en el caso anterior. De esta forma, se dividen en grupos de 4 los datos en las columnas de Q1 y Q0 y se comparan con el biestable en cuestión (T2), para que los valores coincidan en la implementación.

En mi caso, tendría que comparar primero (primera tabla, primer grupo de 4 de T2) con (segunda tabla, primer grupo de 4 de Q1 y Q0):

1
x
1
x

0	0
0	1
1	0
1	1

De esta forma, observo que en la primera salida del decodificador no cambia de ningún modo (es simplemente "1", por lo que sale normal y se une a una puerta OR que finalmente irá a la entrada del biestable). Siguiendo el mismo procedimiento con el resto de las salidas, en la 3º y 4º es 0 (así que no sale nada), pero en la 2º se obtiene Q0' (que sale del biestable J0K0, y se une a una AND junto a la segunda salida del decodificador, y lo que sale de la AND es la segunda entrada de la OR mencionada antes).

	Decodificador 2X4 T2			2^2	
Q3(t)	Q2(t)	Q1(t)	Q0(t)	T2	S
0	0	0	0	1	1
0	0	0	1	x	
0	0	1	0	1	
0	0	1	1	x	
0	1	0	0	1	Q0'
0	1	0	1	0	
0	1	1	0	1	
0	1	1	1	x	
1	0	0	0	x	0
1	0	0	1	x	
1	0	1	0	x	
1	0	1	1	x	
1	1	0	0	x	0
1	1	0	1	x	
1	1	1	0	x	
1	1	1	1	x	

PARA D1: como para D1 lo único especificado era que se necesitaba implementar con puertas lógicas, el procedimiento lógico a seguir es hacer la tabla de Karnaugh de la columna obtenida para D1. Al hacerla, esto es lo que ha salido:

Q3 Q2/Q1 Q0	.00	.01	.11	.10

.00	0	x	x	1
.01	1	0	x	0
.11	x	x	x	x
.10	x	x	x	x

Q2'Q1+Q2Q1'Q0'. Lo único que habría que hacer es multiplicar Q2'(de T2) y Q1(de D1) con una AND, y sumar su salida a la multiplicación (con otra AND) de Q2, Q1' y Q0' (de J0K0). La salida de la OR se une con la entrada del biestable D1, como en el biestable anterior.

PARA J0K0: Para este biestable, se separaría por un lado la entrada J0 y por el otro, K0. Para ambos es necesario implementar un multiplexor 8x1. En el de J0, se sigue un proceso similar al de los decodificadores con los biestables T, pero en este caso, los grupos con los que se comparan son de 2. Primero se eligen las variables de control, que al ser un MUX 8x1, son 3 (Q1, Q2 y Q3. Son las mismas tanto para J0 como para K0). Así se compara lo que aparece a continuación:

Q0(t)	J0
0	1
1	x

Como se observa, la salida de esto sería simplemente 1, por lo que se coloca una constante de 1 en la primera entrada. Se sigue el mismo procedimiento con el resto de las entradas (en todas las demás sale 0, por lo que se conectan las otras 7 entradas a una constante 0). La salida del multiplexor es lo que se une a la entrada de J0.

Multiplexor 8X1 J0 2^3

Q3(t)	Q2(t)	Q1(t)	Q0(t)	J0	E
0	0	0	0	1	1
0	0	0	1	x	
0	0	1	0	0	0
0	0	1	1	x	
0	1	0	0	0	0
0	1	0	1	x	
0	1	1	0	0	0
0	1	1	1	x	
1	0	0	0	x	0
1	0	0	1	x	
1	0	1	0	x	0
1	0	1	1	x	
1	1	0	0	x	0
1	1	0	1	x	
1	1	1	0	x	0
1	1	1	1	x	

Del mismo modo, se sigue el mismo procedimiento para K0. En su caso, lo único que cambia es la posición de los unos y los ceros: la constante 1 se

colocaría en la 3º entrada, y todo lo demás son 0s. La salida del segundo multiplexor entra a la entrada K0.

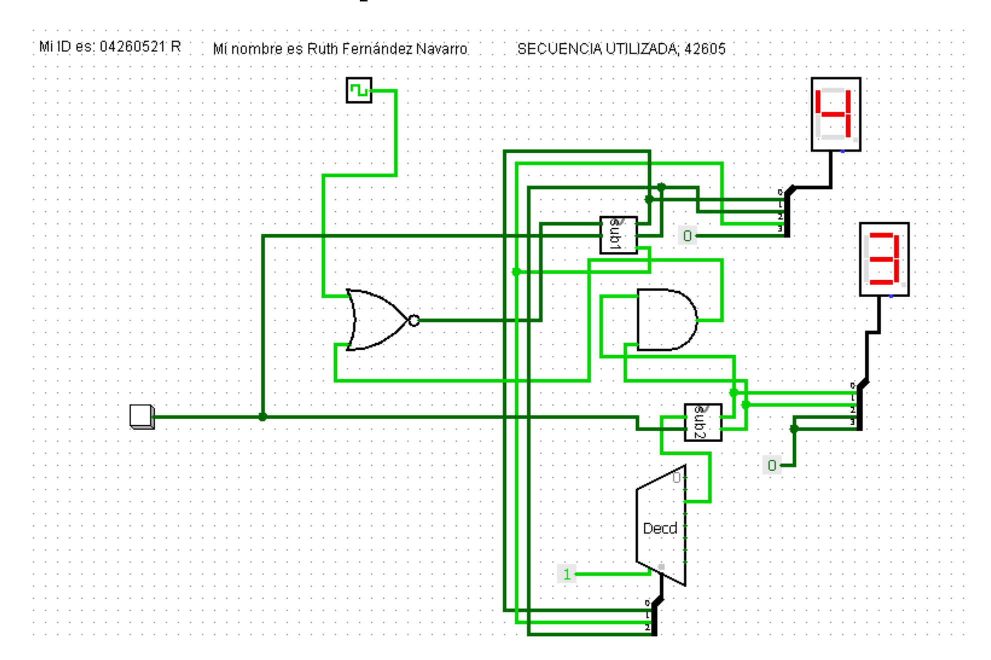
2. SUBCIRCUITO 2.

La función del subcircuito 2 sería lograr que la secuencia del subcircuito 1 se repita 3 veces (con un contador), y que el primer número de la secuencia se mantenga indefinidamente después de las tres repeticiones. A diferencia del subcircuito anterior, no se especifica el uso de biestables o puertas necesarios para este subcircuito, por lo que es a libre interpretación. En mi caso, he hecho uso de dos biestables D, y una puerta AND para unir las salidas de cada uno. La tabla usada es:

Q1	Q0	Q1(t+1)	Q0(t+1)	D1	D0	F
0	0	0	1	0	1	0
0	1	1	0	1	0	0
1	0	1	1	1	1	0
1	1	1	1	1	1	1

3. MAIN

Este sería el main y el resultado final, que consiste en unir los dos subcircuitos, con un botón de reset para poner el contador a 0 y que la secuencia vuelva a empezar:



Mario Sierra Iniesta

Mi secuencia es: 2-3-4-0-9-2-3-4. . . Para ello utilizaré 4 bits para poder representar todos los números de la secuencia.

Al utilizar 4 bits necesitaré 4 biestables siendo estos los que se especifican en la descripción de la tarea T3, T2, D1 y J0 K0.

La tabla a partir de la que se va a crear el circuito es la siguiente:

En la tabla aparecen primero en la izquierda los estados, después están los estados posteriores Q(t+1) y por último los valores de los biestables necesarios para que cada Q cambia al valor que se quiera. Los valores de los biestables nacen de la tabla de excitación de los biestables

Implementación del circuito

Subcircuito 1

El primer subcircuito consiste en realizar un circuito que repita constantemente la secuencia. Sin embargo, hay ciertas condiciones a tener en cuenta a la hora de diseñarlo.

Las condiciones son:

- T3 y T2 han de ser implementados utilizando decodificadores 2x4
- D1 debe ser implementado a través de puertas lógicas
- J0 K0 ha de ser implementado utilizando un multiplexor 8x1

T3: al tener que utilizar un Dec 2x4 significa que hay que decidir cuales van a ser los bits de control, yo he decidido poner Q3 y Q2 como bit de control. Una vez hecho esto hay que dividir la tabla en 4 secciones 1 por cada salida del codificador y por último simplemente hay que fijarse en la columna de T3 de la tabla y hacer que la salida seleccionada coincida con los valores que aparecen en la tabla.

T3	Q3	Q2	Q1	Q0	F	
DECOC 4x2	0	0			1/0	
	0	1			**0**	
	1	0			**1**	
	1	1			**X**	
	Q3	Q2	Q1	Q0	F	
	0	0	0	0	1	**F=NOR Q1 Y Q0**
			0	1	X	
			1	0	0	
			1	1	0	

Por ejemplo, la primera salida del Dec sería cuando Q3 y Q2 valen 0 y para esos valores T3 puede valer 1 o 0 por lo que para poder realizar esto de manera correcta hemos de fijarnos en Q1 y Q0 que he visto que esa parte de la tabla coincide con una puerta NOR entre Q1 y Q0, para ello he conectado esa salida a una puerta AND de dos entradas siendo una de ellas la salida y la otra la salida de la puerta NOR entre Q1 y Q0, posteriormente la salida de la puerta AND se conecta a una puerta OR y la salida de esa puerta OR es la que se conecta a la entrada del biestable, para las posteriores salidas solo habría que repetir este proceso, pero las siguientes salidas solo tienen un posible valor lo que facilita el proceso. En el segundo caso Q3=0 y Q2=1 la salida debe valer 0 por ello simplemente no se conecta a nada esa entrada. En el tercer caso Q3=1 y Q2=0 la salida siempre vale 1 por ello se puede conectar directamente a la puerta OR. De esta manera se implementa el biestable T3 utilizando un Dec 2x4. Lo último sería conectar la salida del biestable a su correspondiente posición del Display siendo este el bit más significativo

T2: el proceso para este biestable es el mismo que para T3 lo único que habría que ajustar sería las salidas del Dec siendo la primera salida la que puede valer 1 o 0 por lo que tocaría fijarse en los dos bits que no sean los de control, que son los mismos que en T3, es decir, los bits de control de T2 son Q3 y Q2, por ende, habrá que fijarse en Q1 y Q0 en este caso se puede

T2	Q3	Q2	Q1	Q0	F	DECOC 4x2	Q3	Q2	Q1	Q0	F		
	0	0			1/0		0	0	0	0	0	F=AND Q1 y Q0	
	0	1			**1**				0	1	X		
	1	0			**0**				1	0	0		
	1	1			**x**				1	1	1		

solucionar haciendo una puerta AND entre Q1 y Q0 y siguiendo el mismo procedimiento de antes

D1: para este biestable se pide implementarlo utilizando solo puertas lógicas, para ello se simplifica su expresión que es la de su columna

D1	Q3Q2/Q1Q0	0.0	0.1	11	10			
	0.0		X		1			
	0.1		X	X	X	Q1(Q0)'	**F=Q3+(Q1(Q0)')**	
	11	X	X	X	X			
	10	X	1	X	X	Q3		

utilizando un mapa de karnaugh

Teniendo ya la expresión simplificada solo se necesitaría conectar la salida de la puerta OR a la entrada del biestable

J0 y K0: para el ultimo biestable se pedía un multiplexor 8x1, para ello se pondrán como bits de control Q3, Q2 y Q1. Para poder hacerlo de manera correcta habrá que hacer 8 divisiones de la tabla correspondiendo cada

una de ellas con una de las entradas del multiplexor, en mi caso este fue fácil de implementar porque K0 valía siempre 1 por lo que no hacía falta pasar por el se podría simplemente conectar una constante de valor 1, en cambio para J0 habría que seguir un procedimiento similar al de los biestables T, fijarse en la columna J0 para saber si debe valer 1 o 0 y en función de eso decidir el valor de la entrada correspondiente del multiplexor

J0	Q3	Q2	Q1	Q0	F	
	0	0	0	0	1	1
	0	0	0	1	X	
	0	0	1	0	1	1
	0	0	1	1	X	
	0	1	0	0	0	0
	0	1	0	1	X	
	0	1	1	0	X	X
	0	1	1	1	X	
	1	0	0	0	X	X
	1	0	0	1	X	
	1	0	1	0	X	X
	1	0	1	1	X	
	1	1	0	0	X	X
	1	1	0	1	X	
	1	1	1	0	X	X
	1	1	1	1	X	

K0	Q3	Q2	Q1	Q0	F	
	0	0	0	0	X	X
	0	0	0	1	X	
	0	0	1	0	X	1
	0	0	1	1	1	
	0	1	0	0	X	X
	0	1	0	1	X	
	0	1	1	0	X	X
	0	1	1	1	X	
	1	0	0	0	X	1
	1	0	0	1	1	
	1	0	1	0	X	X
	1	0	1	1	X	
	1	1	0	0	X	X
	1	1	0	1	X	
	1	1	1	0	X	X
	1	1	1	1	X	

Subcircuito2:

Para el subcircuito 2, el objetivo es diseñar un circuito que identifique las veces que se repite la secuencia, para ello se ha diseñado un contador de 0 a 3 y que una vez llegue 3 se mantenga en ese estado, una vez que este llegue a 3 se activará una puerta AND que parará el subcircuito 1

Q1	Q0	Q1(t+1)	Q0(t+1)	D1	D0	F
0	0	0	1	0	1	0
0	1	1	0	1	0	0
1	0	1	1	1	1	0
1	1	1	1	1	1	1

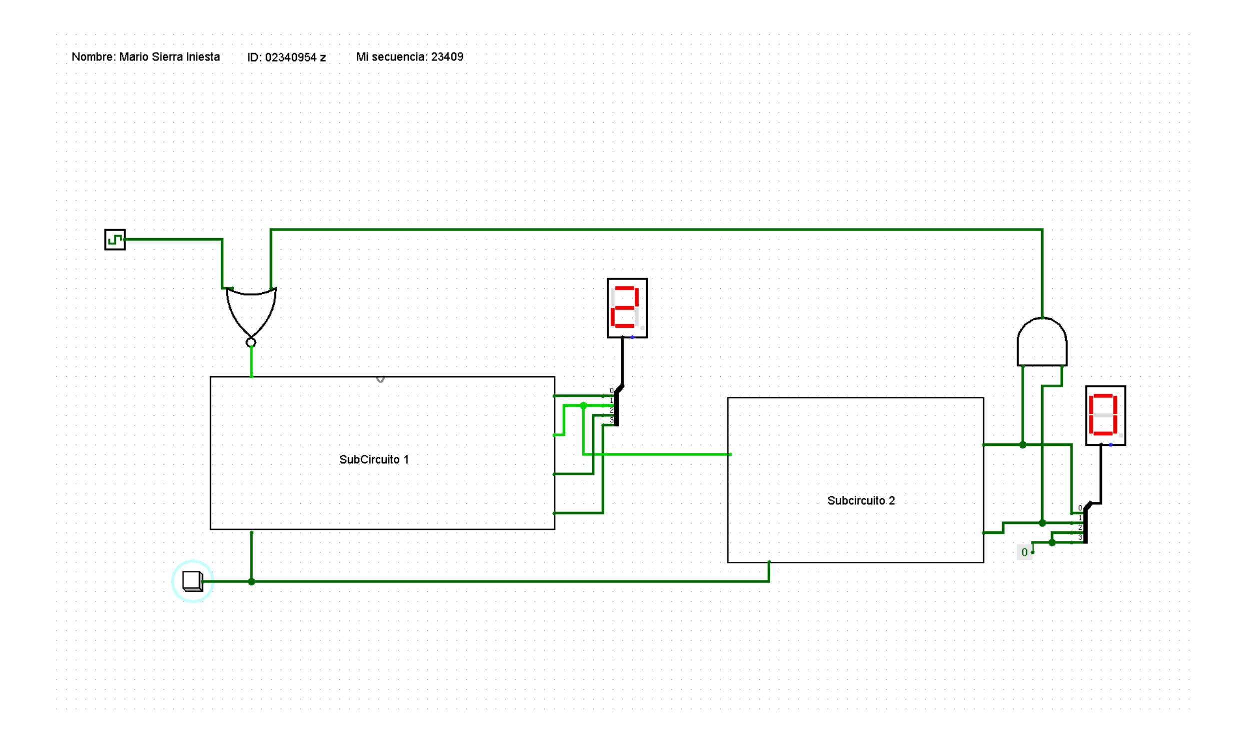
Se ha realizado utilizando dos biestables D y su implementación ha sido a través de la simplificación a través de mapas de Karnaugh de las expresiones de los biestables

D1	Q1/Q0	0	1		
	0		1	Q0	F=Q0+Q1
	1	1	1	Q1	
D0	Q1/Q0	0	1		
	0	1		(Q0)'	F=(Q0)'+Q1
	1	1	1	Q1	

Main:

El main consiste en la combinación de los dos subcircuitos anteriores con la incorporación de dos contadores una para cada subcircuito y botón capaz de reiniciar el funcionamiento de los dos subcircuitos.

Siendo este el resultado



Alejandro Vicioso Oviedo

Secuencia: 2-3-1-9-4-2-3-1-9… Utilizaré 4 bits para poder representar dicha secuencia, que se repetirá 3 veces y cuando se repita 3 veces la secuencia se dejará de ejecutar.

Utilizaré para la representación de esta secuencia unos biestables T3, T2, D1 y JK0, de más significativo a menos significativo respectivamente.

La tabla a partir la que construiré el circuito será la siguiente:

Q3	Q2	Q1	Q0	Q3(t+1)	Q2(t+1)	Q1(t+1)	Q0(t+1)	T3	T2	D1	J0	K0
0	0	0	0	X	X	X	X	X	X	X	X	X
0	0	0	1	1	0	0	1	1	0	0	X	0
0	0	1	0	0	0	1	1	0	0	1	1	X
0	0	1	1	0	0	0	1	0	0	0	X	0
0	1	0	0	0	0	1	0	0	1	1	0	X
0	1	0	1	X	X	X	X	X	X	X	X	X
0	1	1	0	X	X	X	X	X	X	X	X	X
0	1	1	1	X	X	X	X	X	X	X	X	X
1	0	0	0	X	X	X	X	X	X	X	X	X
1	0	0	1	0	1	0	0	1	1	0	X	1
1	0	1	0	X	X	X	X	X	X	X	X	X
1	0	1	1	X	X	X	X	X	X	X	X	X
1	1	0	0	X	X	X	X	X	X	X	X	X
1	1	0	1	X	X	X	X	X	X	X	X	X
1	1	1	0	X	X	X	X	X	X	X	X	X
1	1	1	1	X	X	X	X	X	X	X	X	X

Subcircuito1.

Este subcircuito se encarga de realizar la secuencia con los biestables de manera continua y con la tabla mencionada anteriormente. Para las entradas de los biestables debemos tener en cuenta que deben ser implementadas de la siguiente manera:

T3 y T2 deben ser implementados con un DEC 2X4

D1 debe ser implementado con puertas lógicas

J0 y K0 deben ser implementados con un multiplexor 8X1

A continuación, explicaré como implementé cada entrada.

T3

La tabla que debemos fijarnos para realizar el diseño del decodificador es la siguiente:

Q3	Q2	Q1	Q0	T3
0	0	0	0	X
0	0	0	1	1
0	0	1	0	0
0	0	1	1	0
0	1	0	0	0
0	1	0	1	X
0	1	1	0	X
0	1	1	1	X
1	0	0	0	X
1	0	0	1	1
1	0	1	0	X
1	0	1	1	X
1	1	0	0	X
1	1	0	1	X
1	1	1	0	X
1	1	1	1	X

En este caso nos fijamos donde están los 1s de lo que debe valer T3, en este caso nos beneficia más implementar como variables de control en el decodificador Q0 (como bit menos significativo) y Q1 (como bit más significativo) ya que para el valor 01 la función solo valdrá 1, en el resto de casos nos da igual lo que tenga que valer, ya que tenemos o X o 0, por tanto solo para cuando las variables de control sean Q0 = 0 y Q1 = 1, el decodificador tendrá en su salida 1 un 1 y con eso nos basta.

T2

La tabla que debemos fijarnos para realizar el diseño del decodificador es

Q3	Q2	Q1	Q0	T2
0	0	0	0	X
0	0	0	1	0
0	0	1	0	0
0	0	1	1	0
0	1	0	0	1
0	1	0	1	X
0	1	1	0	X
0	1	1	1	X
1	0	0	0	X
1	0	0	1	1
1	0	1	0	X
1	0	1	1	X
1	1	0	0	X
1	1	0	1	X
1	1	1	0	X
1	1	1	1	X

la siguiente:

En este caso utilicé como variables de control Q3(más significativo) y Q2 (menos significativo). Si dividimos la tabla en 4 partes, y si nos fijamos, para cuando la tabla es 01 o 10, podemos decir que T2 vale 1 en su totalidad ya que tenemos un 1 en 0100 y 1001 y el resto son X por tanto podemos conectar con una puerta OR las salidas 1 y 2 del decodificador a la entrada T2.

D1

En este caso nos vamos en fijar en la siguiente tabla:

Q3	Q2	Q1	Q0	D1
0	0	0	0	X
0	0	0	1	0
0	0	1	0	1
0	0	1	1	0
0	1	0	0	1
0	1	0	1	X
0	1	1	0	X
0	1	1	1	X
1	0	0	0	X
1	0	0	1	0
1	0	1	0	X
1	0	1	1	X
1	1	0	0	X
1	1	0	1	X
1	1	1	0	X
1	1	1	1	X

En este caso debemos usar el método de simplificación con mapas de Karnaugh, la tabla resultante es la siguiente:

Q3Q2\ Q1Q0	0.0	0.1	1.1	1.0
0.0	x	0	0	1
0.1	1	x	x	x
1.1	x	x	x	x
1.0	x	0	x	x

Con las simplificaciones debidamente hechas, se nos queda en que D1 = Q2 + Q1Q0'

Vendría siendo una puerta AND entre Q1 y Q0' y una puerta OR conectada a la AND y a Q2

J0 y K0

La tabla en la que nos fijaremos para implementar los multiplexores 8X1 será la siguiente:

Q3	Q2	Q1	Q0	J0	K0
0	0	0	0	X	X
0	0	0	1	X	0
0	0	1	0	1	X
0	0	1	1	X	0
0	1	0	0	0	X
0	1	0	1	X	X
0	1	1	0	X	X
0	1	1	1	X	X
1	0	0	0	X	X
1	0	0	1	X	1
1	0	1	0	X	X
1	0	1	1	X	X
1	1	0	0	X	X
1	1	0	1	X	X
1	1	1	0	X	X
1	1	1	1	X	X

Para J0 la implementación será tan simple como dividir la tabla en 8 trozos, poner como variables de control a Q3, Q2 y Q1 y debemos fijarnos en 2 trozos, el segundo y el tercero ya que aunque tengamos X, hay un 1 en el segundo trozo y un 0 en el tercero, por tanto debemos conectar a corriente la entrada 1 del multiplexor y a tierra la conexión 2 del multiplexor, el resto debemos conectarlo a tierra o a corriente, es indiferente ya que son todos X

Para K0 la implementación será similar, dividimos la tabla en 8 trozos y ponemos de variables de control a Q3, Q2 y Q1, y nos debemos de fijar en el trozo 1, 2 y 5. Los trozos 1 y 2 deben ir conectados a tierra si o si y el trozo 5 a corriente, el resto es indiferente si va conectadoa tierra o a corriente ya que son todo X.

Subcircuito 2

El subcircuito 2 se encarga de realizar el conteo de veces que se ha repetido la secuencia, se necesitan 2 bits para que cuente de 0 a 3

Utilice 2 biestables tipo D y ambos los implemente con puertas lógicas con simplificación a través de mapas de Karnaugh

Nuestra tabla principal simplificada será la siguiente:

E	Q1	Q0	Q1(t+1)	Q0(t+1)	D1	D0
1	0	0	0	1	0	1
1	0	1	1	0	1	0
1	1	0	1	1	1	1

E es cuando el circuito detecta que hay un 2, esto se verá en el main

Los mapas de Karnaugh resultantes son los siguientes:

E\Q1Q0	0.0	0.1	1.1	1.0
0	x	x	x	x
1	1	0	x	1

D0 = Q0’

E\Q1Q0	0.0	0.1	1.1	1.0
0	x	x	x	x
1	0	1	x	1

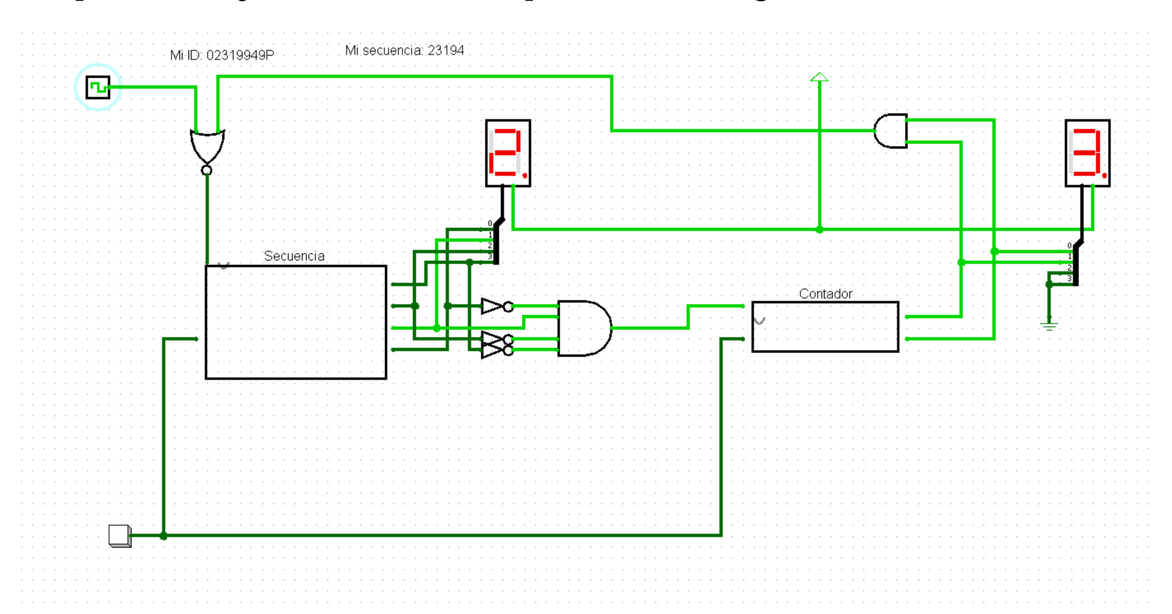
D1 = Q0 + Q1

D0 se implementará conectando únicamente a la puerta Q0'

D1 será una puerta OR de Q0 y Q1

Main

Ponemos los 2 subcircuitos en forma de chip y un reloj conectado al subcircuito 1, el subcircuito 2 funcionará de forma asíncrona cuando haya un 2 en la secuencia. detectado por una puerta AND. Para que la secuencia deje de hacerse, utilizaremos una puerta AND que nos diga cuando el circuito ha llegado a 3 y una NOR para que cuando esa AND sea 1, se bloquee el reloj. El circuito main quedará de la siguiente forma:



Link al vídeo

https://youtu.be/3weRZyqouJQ

Bibliografía

Nos hemos apoyado en los temas dados durante la asignatura, así como de los contenidos vistos en el laboratorio.